**One-sided t-test(aj 2-sample)**

* testuje ci je stredna hodnota vacsia/mensia ako x
* 2-sample: testuje, ci je jedna stredna hodnota vacsia/mensia ako druha
* Alternative: less/greater
* H0 -Velocity<=792 H1- Velocity>792
* t.test( Velocity ,alternative="greater" , mu=792 )
* qt(0.95, n-1)

**Two-sided t-test (aj 2-sample)**

* Testuje nerovnost vs rovnost strednej hodnoty a konkretnej value
* 2-sample: nerovnost vs rovnost dvoch strednych hodnot (z dvoch roznych sad dat)
* Alternative: two-sided
* H0 - Velocity sa rovna 792, H1- Velocity sa nerovna 792
* t.test( Velocity , alternative="two.sided" , mu=792 )
* Kriticka hodnota
* qt(0.95, n-1)
* Rucny vypocet p-value
* pt(t, n-1)

**Variancia - predtest:**

* Testujeme pre dve sady samples, ci maju rovnakú variance(rozptyl)
* var.test(Velocity[Trial=="first"], Velocity[Trial=="second"])
* H0: var sa zhodujú, H1: var sa nezhodujú

**Welsch:s**

* Dve nezavisle sady normalnych dat, ktore nemaju rovnaky variance
* t.test(PT1,PT2,  
   alternative="two.sided",  
   var.equal=FALSE)

**Studentov test:**

* Pre parove testy alebo ak variance su rovnake, moze byt two-sided, one-sided...

**Parovy t-test:**

* 2-sample test, ked su dve sady dat závislé
* Testujeme H0: mean(materialA) <= mean(materialB) vs. H1: mean(materialA) > mean(materialB)
* t.test(materialA , materialB ,  
   alternative="greater",  
   paired=TRUE)

**Kolmogorov-Smirnov test:**

* Testuje, ci su data z normálneho rozdelenia
* Testujeme H0: data su z norm. rozdelenia vs. H1: data nie su z norm rozdelenia
* ks.test( Velocity ,  
   "pnorm" ,  
   mean=mean(Velocity) ,  
   sd=sqrt(var(Velocity)) )

**Shapirov-wilkov**:

* Testuje normalitu pre malu sadu dat, napr. 10
* H0: data z normalneho rozdelenia vs. H1: data nie su z normalneho rozdelenia
* shapiro.test( materialA - materialB )

**Wilcoxon:**

* wilcox.test(PT1,alternative="greater",mu=0)
* H0: mean(PT1) <= 0 vs. H1: mean(PT1) > 0
* wilcox.test(PT1,PT2, alternative="two.sided")

**Kritická hodnota:**

* Pre akú vstupnú hodnotu je p-value 5%?
* Df - stupen volnosti daneho rozdelenia

qt(0.95, df = length(rozdiely)-1)

## Rozhodovanie:

1. paired/not paired
2. normal/not normal
3. Variance = true/false
4. One -sided -two-sided

Jedna sada dat:

1. One-sided vs two-sided
2. norm/nenorm rozdelenie
   1. Normalne rozd. -> Studentov
   2. Nenormalne rozd. -> Wilcoxon

Dve sady dat:

1. One-sided vs two-sided
2. zavisle/nezavisle
   1. Zavisle (paired=TRUE):
      1. Normalita rozdielov
         1. Nenormal -> **Wilcox**, (netestujem uz variance, paired=TRUE)
         2. Normal -> **parovy** t-test (paired=TRUE)
   2. Nezavisle:
      1. Normal:
         1. Test variancii (predtest):
            1. Rovnost: **Studentov** (var.equal=TRUE)
            2. Nerovnost: **Welschov** (var.equal=FALSE)
      2. Nenormal:
         1. Nenormal -> **Wilcox**, (netestujem uz variance, paired=FALSE)

Kriticka hodnota

\alpha kriticka hodnota = 1 - \alpha kvantil (hranica testovej statistiky vyznamnosti)

Pozriem sa v t.teste na testovu statistiku a ked je mensia, tak je v tom zlom pocte dat

V 2 sided sa pozriem, ci je hodnota medzi - a + \alpha kritickou hodnotou a ak hej, tak je opat v zlom pocte dat a potvrdzujem H0

Outliery:

Motivacia: vyhadzujeme data, ktore su prilis mimo rozdelenia, na boxplote sa zobrazia az za hradbami

**Vypocet**:

* Najdeme hranice 1. A 3. Kvantilu: quantile(novy, probs = 0.25), quantile(novy, probs = 0.75),
* Zratame IQR(Interquartile range): 3.kvartil - 1. Kvartil
* Na zaklade IQR zratame dolnu hradbu: 1. Kvartil - 1.5 IQR a hornu hradbu: 3. Kvartil + 1.5 IQR (tento vzorec je v tahaku)
* Vyhodime vsetko co je mensie ako spodna hradba a vacsie ako hrona hradba

remove.outliers <- function(data){

IQR <- quantile(data, probs = 3/4) - quantile(data, probs = 1/4)

dolna <- quantile(data, probs = 1/4) - 1.5 \* IQR

horna <- quantile(data, probs = 3/4) + 1.5 \* IQR

return(data[data >= dolna & data <= horna])

}

Sklon dát testy:

* Nakreslíme si qqnorm(data) - Pozitivny sklon - U, Negativny sklon - Opacne U, Normalny sklon - priamka
* Zratame skewness (vzorec je v tahaku):
  + skewness < 0 -> neg. Sklon
  + Skewness == 0 -> norm. Data
  + Skewness > 0 -> pos. Sklon
* Zratame kurtosis (vzorec - tahak) vydutost dat, ako velmi spicaty je kopcek rozdelenia :) :
  + K > 3 -> spicaty kopec, tazke chvosty
  + K = 3 -> norm. Rozd.
  + K < mohutny kopec, “lahke” chvosty

# Testy významnosti korelácie:

**Pearson:**

* Počítame korelačný koeficient \in <-1, 1>
* H0: rho <=0 vs. H1: rho > 0
* cor.test(reklama, trzba, alternative = "greater", method = "pearson")

**Spearman:**

**Fischerova Z-premenná:**

* IS pre rho, ak sú dáta z N(,)
* Z = atanh(cor(X, Y))
* dolna hranica:
* tanh( Z - qnorm(0.975) / sqrt(length(X)-3) )
* horna hranica:
* tanh( Z + qnorm(0.975) / sqrt(length(X)-3) )

**2-sample test:**

## Rozhodovanie:

normal/not normal

1. Normal -> Pearson
2. Not normal -> Spearman